A novel model for hourly $PM_{2.5}$ concentration prediction based on CART and EELM

Zhigen Shang[a*], Tong Deng[b], Jianqiang He[a], Xiaohui Duan[a]

[a] *Department of Automation, Yancheng Institute of Technology, Yancheng 224051, China*

[b] *The Wolfson Centre for Bulk Solids Handling Technology, Faculty of Engineering & Science, University of Greenwich, Kent, ME4 4TB, UK*

ABSTRACT

Hourly $PM_{2.5}$ concentrations have multiple change patterns. For hourly $PM_{2.5}$ concentration prediction, it is beneficial to split the whole dataset into several subsets with similar properties and to train a local prediction model for each subset. However, the methods based on local models need to solve the global-local duality. In this study, a novel prediction model based on classification and regression tree (CART) and ensemble extreme learning machine (EELM) methods is developed to split the dataset into subsets in a hierarchical fashion and build a prediction model for each leaf. Firstly, CART is used to split the dataset by constructing a shallow hierarchical regression tree. Then at each node of the tree, EELM models are built using the training samples of the node, and hidden neuron numbers are selected to minimize validation errors respectively on the leaves of a sub-tree that takes the node as the root. Finally, for each leaf of the tree, a global and several local EELMs on the path from the root to the leaf are compared, and the one with the smallest validation error on the leaf is chosen. The meteorological data of Yancheng urban area and the air pollutant concentration data from City Monitoring Centre are used to evaluate the method developed. The experimental results demonstrate that the method developed addresses the global-local duality, having better performance than global models including random forest (RF), *v*-support vector regression (*v*-SVR) and EELM, and other local models based on season and *k*-means clustering. The new model has improved the capability of treating multiple change patterns.

Keywords: $PM_{2.5}$ concentration prediction; Local model; Classification and regression tree (CART); Extreme learning machine (ELM); Ensemble model

1. Introduction

Predicting concentrations of particulate matter in the air is important for control and reduction of airborne pollution. Particulate matter refers to small particles consisting of dust, dirt, soot, smoke, and liquid droplets suspended in the air. These particles vary widely in size (aerodynamic diameter). Particles with an aerodynamic diameter less than 2.5μm ($PM_{2.5}$) are known as fine particulates, which are seriously harmful to human health because of its absorption of toxic substances such as carcinogenic organic compounds and heavy metals (Thomaidis et al., 2003; Zhou et al., 2014). Exposure to high concentrations of $PM_{2.5}$ has been linked to many diseases, such as diabetes, lung cancer, respiratory

*Corresponding author at: Department of Automation, Yancheng Institute of Technology, Yancheng 224051, China.
E-mail address: zgshang@ycit.edu.cn (Z. Shang).

and cardiovascular diseases (Requia et al., 2017). It is thought that prediction of $PM_{2.5}$ concentrations is beneficial to improve early warning procedures.

In recent years, a variety of models have been developed to predict $PM_{2.5}$ concentrations (Niu et al., 2016; Wang et al., 2017). These models are basically classified into two categories: deterministic and statistical models. Deterministic models, called chemical transport models (CTMs) focus on understanding the underlying complex interactions among meteorology, chemistry and emission. CTMs simulate the atmospheric chemistry and physics in the emission, transport and transformation processes. Thus, CTMs require sufficient information of pollutant sources, reaction mechanisms and chemical kinetics and so on (Sun et al., 2013). Insufficient information of pollutant sources and improper representation of physicochemical processes limit their application in many places (Qin et al., 2014). Another disadvantage of CTM approaches is high computational cost (Doraiswamy et al., 2010). On the other hand, statistical models aim to develop the relationships between selected input variables and air pollutant concentrations using various regression models. These models usually need a sufficient amount of historical data from monitoring stations. Compared to deterministic models, statistical models have the advantage of easy, quick and economical implementation when given sufficient historical data (Wang et al., 2017). Furthermore, statistical approaches are generally more competent in capturing the underlying site-specific dependencies between air pollutant concentrations and selected variables. Therefore, statistical models are less complex, computationally intensive but more accurate (Perez, 2012).

Statistical models mainly include linear and generalized linear regression, nonlinear regression, autoregressive integrated moving average (ARIMA), hidden Markov model (HMM), random forest (RF), support vector regression (SVR) and artificial neural network (ANN). Vlachogianni et al. (2011) adopted the linear regression model to predict NO_x and PM_{10} concentrations using NO, NO_2, CO, O_3 and $PM_{2.5}$ concentrations. The generalized linear model was used to predict PM_{10} concentrations in urban areas (Garcia et al., 2016). Cobourn (2010) presented a $PM_{2.5}$ prediction model based on nonlinear regression and back-trajectory $PM_{2.5}$ concentrations. However, the methods based on linear, generalized linear and nonlinear regression tend to oversimplify the relationships between air pollutant concentrations and predictor variables. Ni et al. (2017) designed an ARIMA time series model to explore the prediction of $PM_{2.5}$ in the short-term time series. However, the ARIMA model, being a linear model, cannot be well adapted to the nonlinear air pollutant series (Niu et al., 2016). Dong et al. (2009) developed a method based on the hidden semi-Markov model to predict $PM_{2.5}$ concentration levels. Moreover, Sun et al. (2013) proposed an HMM with different emission distributions to predict 24-hour-average $PM_{2.5}$ concentrations in Northern California. The HMM model, however, suffers from several inherent shortcomings, such as computationally expensive training and its sensitivity to initial condition (Budalakoti et al., 2009). Random forest is a substantial modification of bagging that builds a large collection of de-correlated trees, and then averages them (Breiman, 2001). Yu et al. (2016) used the random forest algorithm to

predict air quality for urban sensing systems. Support vector regression (SVR) is based on structural risk minimization, and it has a unique global optimal solution. Kernel function applied in SVR maps the original space into a high-dimensional feature space, where a linear regression model fits the data more appropriately. The SVR is widely utilized to predict $PM_{2.5}$ and other pollutant concentrations (Lu and Wang, 2005; Niu et al., 2016; Xu et al., 2017). However, several model parameters in SVR play a critical role in prediction performance. SVR requires considerable computational cost to fine-tune the parameters. The ANN models have also gained popularity in their use for the prediction of $PM_{2.5}$ concentrations due to the capability of detecting complex underlying nonlinear relationships (McKendry, 2002; Qin et al., 2014; Voukantsis et al., 2011). However, the results of an ANN model are sensitive to the selection of model parameters, and the model requires considerable expertise for fine-tuning the parameters.

Extreme learning machine (ELM), proposed by Huang et al. (2006), is an efficient learning algorithm for single-hidden layer feedforward neural networks (SLFNs). In ELM, the parameters of the hidden layer are randomly initialized, and the weights of the output layer are analytically computed by using Moore-Penrose generalized inverse. Thus, ELM model can be run in an extremely low computational time. However, Moore-Penrose generalized inverse leads ELM to suffer from the overfitting problem. Wang et al. (2017) utilized differential evolution (DE) to optimize the parameters of the hidden layer for predicting air quality index. In addition to optimization algorithm, ensemble algorithm can also be used to improve the performance of ELM (Zhou et al., 2002).

The selection of input variables for an ANN based prediction model is a critical issue, since irrelevant or noisy variables may result to an unnecessarily complex model and poor generalization (Voukantsis et al., 2011). The linear framework, including Pearson correlation coefficient, multiple linear regression and forward stepwise regression and so on, has been widely used to select a reduced set of input variables (Cobourn, 2010; Díaz-Roles et al., 2008; Ni et al., 2017; Russo et al., 2013). However, the linear framework has the limited capability of stimulating the complex relationships between air pollutant concentrations and input variables. Voukantsis et al. (2011) employed a combination of linear regression and ANN to select input variables for air pollutant prediction. To be completely free from the usual linear framework, Poggi and Portier (2011) used the RF variable importance to determine input variables for PM_{10} prediction. In the RF framework, the most widely used measure of importance of a given variable is the mean difference of prediction errors of the trees (mean squared error (MSE) for regression) before and after the values of this variable are randomly permuted in the out-of-bag (OOB) samples (Poggi and Portier, 2011). If a variable is important, a random permutation will degrade the prediction, and the mean difference will be a large value.

Due to seasonal variations and diurnal variations, $PM_{2.5}$ concentrations have multiple change patterns (He et al., 2017; Kassomenos et al., 2014; Liu et al., 2015). McKendry (2002) suggested that for air pollution prediction, hybrid models and local multi-layer perceptions (MLPs) may be superior to a single global MLP. Hybrid models try to

construct a set of prediction models and combine them. Díaz-Roles et al. (2008) constructed a hybrid model, aggregating the outputs of ARIMA and ANN to capture different patterns in PM_{10} concentrations. The hybrid model developed by Perez (2012) used nearest neighbour model as a correction which was applied to the results obtained from the ANN model. Recently, hybrid models based on time series decomposition have been gradually developed, which mainly consist of the following three steps: (1) decompose the output time series into several components; (2) train a prediction model for each component; (3) aggregate the outputs from different component models. For air pollutant, some hybrid prediction models apply wavelet transform as a tool to decompose output time series (Bai et al., 2016; Osowski and Garanty, 2007). Additionally empirical mode decomposition (EMD) and its variations also have been used to decompose the original time series (Niu et al., 2016; Wang et al., 2017; Xu et al., 2017; Zhou et al., 2014; Zhu et al., 2017).

Compared to hybrid models, local models split the input space into subspaces with similar properties and construct a prediction model for each subspace. In this study, global models are referred as single models developed on the total training data wherein air pollutant concentrations are from an air quality monitoring station, but local models are trained on the subsets of the training data used for global models. The local models based on season divide the training set into seasonal subsets, and train a model for individual season. Perez and Gramsch (2016) developed an ANN model to predict hourly $PM_{2.5}$ in Santiago de Chile for the season when high concentration episodes occur frequently, with the training data from the same season for years 2010 and 2011. To improve air pollutant prediction, Feng et al. (2011) used clustering method to divide the dataset into several clusters and built an ANN for individual cluster. However, the output was ignored when using clustering algorithm to split the dataset, which was unreasonable. Moreover, Bettenburg et al. (2012) found that balancing thinking locally and acting globally is important for local models. Normally global models have the risk of underfitting due to multiple patterns, while local models tend to suffer from the overfitting problem. Furthermore, a global model may be beneficial for learning some patterns. Consequently, air pollutant prediction based on local models needs to address the global-local duality.

This study develops a framework based on CART and EELM to deal with the global-local duality. Through constructing a shallow regression tree by using CART, the whole dataset is divided into subsets in a hierarchical manner. For each node of the tree, the EELMs are trained using the samples belonging to the node, and hidden node numbers are selected to minimize validation errors respectively on the leaves of a sub-tree that takes the node as the root. For each leaf, there are a global and several local EELMs on the path from the root node to the leaf, and the EELM with the smallest validation error on the leaf is selected. Before the implementation of the CART-EELM, input variables are selected by using the RF model. The meteorological data of Yancheng urban area and the air pollutant concentration data from City Monitoring Centre are used to evaluate the capability of the CART-EELM model in dealing with the

global-local duality.

The rest of this paper is organized as follows: CART and EELM are introduced, and CART-EELM is further proposed in Section 2. In Section 3, our CART-EELM is evaluated with comparison to several other models. Section 4 draws the final conclusions.

2. Methodology

2.1. Selection of input variables using RF

In Poggi and Portier (2011), input variables were selected for PM_{10} prediction through the analysis of the RF variable importance. In this study, the RF model is also employed to provide variable importance ranking, but the variable selection is performed based on the cross-validation error of the RF model. The selection of input variables consists of the following steps:

Step1: Assess the OOB error of the RF model, compute the importance scores of input variable candidates, and rank the candidates in a descending order of importance. To minimize sampling effects, we run the RF ten times on the training set, and the importance score of each candidate is the mean of the scores observed from ten RF models.

Step 2: Invoke the most important k variables at the beginning, implement sequential introduction, and use the cross-validation error of the RF model to evaluate the different combinations of input variable candidates. The combination with the lowest error is chosen.

Step 3: Return to Step 1 until no further candidates can be rejected.

2.2. CART

CART, developed by Breiman et al. (1984), explores the structure of the training set and generates easily understandable decision rules for regression or classification. The basic idea of the algorithm is to recursively partition the input space into binary subsets where the output becomes successively more homogeneous.

Let $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{l}$ be a set of training samples, where $\boldsymbol{x}_i \in \mathbf{R}^m$ is the ith input vector and $y_i \in \mathbf{R}$ is the corresponding output. CART begins with the root node, which is associated with the entire input space $\mathbf{R}^m$. The root node contains all the training samples. The next step is calculating the first split. For a regression problem, the split is to minimize the expected sum variances for two resulting subsets:

$$\begin{aligned} &\min_{j,c} \frac{1}{l}\left(\sum_{k\in S_L} (y_k - \overline{y}_L)^2 + \sum_{k\in S_R} (y_k - \overline{y}_R)^2 \right) \\ &\text{s.t. } S_L = \left\{ i \middle| \boldsymbol{x}_{ij} \le c, i = 1,\ldots l \right\}, \\ &\quad S_R = \left\{ i \middle| \boldsymbol{x}_{ij} > c, i = 1,\ldots l \right\}, \\ &\quad j \in \{1,\ldots,m\} \end{aligned} \tag{1}$$

where S_L and S_R are the sets of training indices going to left child node and right child node, $\overline{y}_L$ and $\overline{y}_R$ are the mean

values of the outputs of samples in two subsets. The optimal j and c can be easily determined by discrete search over the m input dimensions and l samples. The children of the root node are recursively split in the same manner until some stop criterion is satisfied.

CART has low computational complexity because of its recursive computation. By moving from the root node to the terminal node (leaf), each example is then assigned to a unique leaf where the predicted value is determined. Normally, the mean value of the outputs of samples fallen into the leaf is chosen as the predicted value. So CART is nonparametric and can find complex relationships between input and output variables. Therefore, CART also has the advantage of discovering nonlinear structures and variables interactions in the training samples (Brezigar-Masten and Masten, 2012).

Since the split in CART aims to minimize the diversity of outputs, the model is a natural fit for pattern segmentation. In this study, the results of CART are used to replace k-means clustering, where CART is used to segment the change patterns of hourly $PM_{2.5}$ concentrations. The k-means clustering is a kind of unsupervised learning, while CART is a form of supervised learning. So CART has the advantage of considering the output when segmenting patterns.

2.3. EELM

In ELM, the hidden layer parameters are randomly initialized. ELM is mathematically modelled by

$$o_i = \boldsymbol{g}(\boldsymbol{x}_i)^{\mathrm{T}} \boldsymbol{\mu}, \tag{2}$$

where $\boldsymbol{g}(\boldsymbol{x}_i) \in \mathbf{R}^p$ is the output vector of the hidden layer, $\boldsymbol{\mu} \in \mathbf{R}^p$ is the output weight vector, and p is the number of the hidden neurons. The RBF nodes are used in this study, therefore Eq. (2) is rewritten as

$$o_i = \sum_{j=1}^{p} \mu_j \exp(-\left\| \boldsymbol{x}_i - \boldsymbol{a}_j \right\|^2 / 2b_j^2), \tag{3}$$

where $\boldsymbol{a}_j \in \mathbf{R}^m$ and $b_j \in R^+$ are the centre and impact factor of jth RBF node. ELM computes the output weight vector $\boldsymbol{\mu} = \boldsymbol{H}^{\dagger} \boldsymbol{y}$ by Moore-Penrose generalized inverse, where $\boldsymbol{H} = [g(\boldsymbol{x}_1), \ldots, g(\boldsymbol{x}_l)]^{\mathrm{T}}$, and $\boldsymbol{y} = [y_1, \ldots, y_l]^{\mathrm{T}}$. Thus, the ELM has an extremely low computational time. However, Moore-Penrose generalized inverse leads ELM to suffer from the overfitting problem.

Ensemble algorithm is one of methods to improve the generalization performance of ELM. Several practical works showed that the performance of a single ELM can be improved by using properly ensemble techniques, which develop a population of ELM-based learners and then combine them to create improved results (Xue et al., 2014).

Bagging and boosting are typical representatives of ensemble methods. Khoshgoftaar et al. (2011) proved that bagging generally outperforms boosting. Xue et al. (2014) proposed a novel ensemble of extreme learning machine based on Genetic algorithms (GE-ELM), which outperforms both bagging and boosting. However, GE-ELM needs

considerably more training time. It is worth noting that in this study, any ensemble methods can be used. By considering performance and computational time, bagging is chosen to ensemble ELMs. If it is assumed that all ELMs in EELM have the same number of hidden neurons, there is only one parameter in the model, hidden neuron number, which needs to be optimized when using EELM.

2.4. CART-EELM model

For prediction of hourly $PM_{2.5}$ concentration, a novel method on training local models based on combination of CART and EELM is developed, which aims at addressing the global-local duality and improving the prediction accuracy. The algorithm developed includes the following steps:

Step 1: Construct the CART tree using the training set. The deeper branches in the tree may be affected by outliers. Moreover, local models are trained at each non-root node of the tree in our algorithm. Thus, a shallow tree is constructed to capture concentration change patterns and to ensure that each leaf has enough training samples for its local model. To generate a shallow tree, a large value of the minimum number of samples in a leaf is set. Furthermore, considering the fact that samples with the low-value outputs may take the most of the dataset, the maximum depth of the tree is also set to prevent these samples from being excessively split. A CART tree is constructed using the total training samples, and is then applied for splitting the validation samples. So each node has its own training and validation samples.

Step 2: Train EELMs using the associated samples. Each node in the hierarchical tree trains EELMs using the associated training samples of the node. At the root node, global models use the total training set, but utilize different validation subsets from the leaves of the tree to respectively determine the model parameters. At each internal node, local EELMs are trained with its own training samples, and the model parameters are chosen to minimize the validation errors respectively on the leaves of a sub-tree that takes the node as the root. At each leaf node, a local EELM is obtained with its own training and validation samples.

Step 3: Compare a global and local EELMs associated to the leaf. For each leaf of the tree, a global and several local EELMs on the path from the root node to the leaf are compared, and the one with the minimum validation error on the leaf is chosen.

Given a testing sample, it is assigned into a unique leaf where its prediction model is determined. The testing procedure consists of two steps:

Step1: Assign the testing sample to a unique leaf using the splitting rules of the developed tree.

Step2: Make a predicted value for the testing sample using the prediction model chosen for the assigned leaf.

Local models at deep levels (e.g., at the leaves) may suffer from the overfitting problem. A global model trained on the total dataset is hard to learn well all the patterns, but may be beneficial for capturing some patterns. Firstly, the

prediction model for a leaf is chosen based on the comparison of a global model and local models on the path from the root node to the leaf, which can address the global-local duality. For the testing samples in a leaf (an input subspace), training a prediction model in a larger input space and selecting the model parameter to minimize the validation error on the leaf samples may be beneficial. The CART-EELM model aims to seek the best training set for each subspace (i.e., each leaf) based on the hierarchical regression tree. Secondly, at each non-leaf node, we choose different model parameters for different leaves of a sub-tree that takes the node as the root node, since different leaves may have distinct patterns. Finally, benefitting from high computational efficiency of EELM, the CART-EELM model has acceptable computational cost.

3. Data and model implementation

3.1. Data

The experiments on a real-world air pollution dataset are made to prove the effectiveness of the proposed model. The dataset is from Yancheng city, which is one of the 13 cities under the direct administration of Jiangsu Province, China. Yancheng city spans between northern latitude 32°34'-34°28', eastern longitude 119°27'-120°54'. As shown in Fig. 1, the city is located in the middle of the northern plain of Jiangsu, and on the east of the city is the Yellow Sea. Yancheng is fast growing urbanization. Increased energy consumption and the number of vehicles prompt the air pollutant exposure levels especially for the fine particulates. Although clean energy and green technologies are encouraged in Yancheng and neighbouring cities, this city still has experienced severe pollution events.

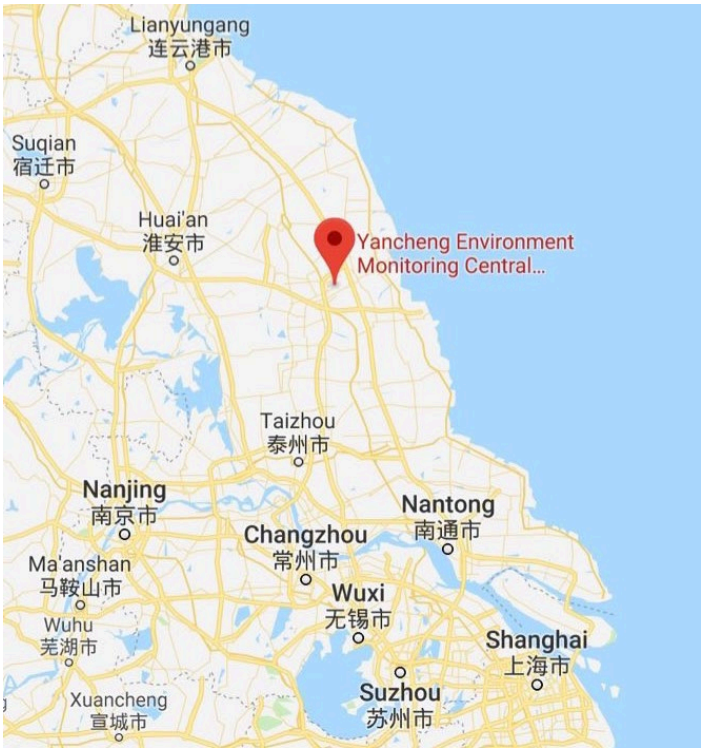

Fig. 1. Location of Yancheng Environment Monitoring Central Station

To monitor air pollution, networks of monitoring stations in major cities of China have been established. The hourly average concentrations of six air pollutants ($PM_{2.5}$, O_3, PM_{10}, SO_2, NO_2, CO) are automatically recorded at monitoring stations. In Yancheng urban district, there are five monitoring stations: Wenfeng Middle School, Yancheng Power Plant, Administrative Committee of Development Zone, Baolong Square and City Monitoring Centre.

Since early 2013, the Environmental Protection Administration began to publish these real-time data to the public. Now the data from the station of City Monitoring Centre are used in this study, which is located in an area with the high population density. The air quality data encompasses July 1, 2015 to June 30, 2018. In addition, the meteorological data of Yancheng urban area from July 1, 2015 to June 30, 2018, which consists of surface wind speed (WS) and direction (WD), temperature (T), surface air relative humidity (RH) and rainfall (R), are also incorporated into the models. The data from July 1, 2015 to December 31, 2017 are used for training; the data from January 1, 2018 to June 30, 2018 are used for testing. The models involved are developed in Python 3.6 using Anaconda 3. The experiments are conducted on a computer with a Win10 64 bit OS running on Intel Core i5-7440HQ with 8 GB RAM.

Some values in the air quality and meteorological data are missing for the studied period. The missing values are interpolated by using cubic spline interpolation when the missing gap is less than 3 hours and there is no missing value in 3 hours before and after. Let the observed $PM_{2.5}$ value at time t+1, denoted as $PM_{2.5}(t+1)$, be the output of a sample. The corresponding input candidates include air pollutants at times t-2, t-1 and t, and the meteorological data at time t. Moreover, the meteorological data at time t+1 is also included as input candidates, since the predicted meteorological data for the next hour can be used in practical implementation. The -sine and cosine transformations are employed for the wind direction (Feng et al, 2015). Weekend indicator (1 if yes, 0 otherwise) for time t+1, abbreviated WEI(t+1), is added due to alterations in traffic patterns and industrial behaviours at weekends. The corresponding input candidates are given in Table 1. After the samples with any missing value are deleted, we have 19627 training samples and 3898 testing samples.

Table 1: The input variable candidates corresponding to the output $PM_{2.5}(t+1)$.

Input candidate names	Input candidate vector
$XC_0,\ldots,XC_{17}$	$CO(t-2),CO(t-1),CO(t),NO_2(t-2),NO_2(t-1),NO_2(t),$ $O_3(t-2),O_3(t-1),O_3(t),SO_2(t-2),SO_2(t-1),SO_2(t),$ $PM_{10}(t-2),PM_{10}(t-1),PM_{10}(t),PM_{2.5}(t-2),PM_{2.5}(t-1),PM_{2.5}(t)$
$XC_{18},\ldots, XC_{29}$	$T(t)$, $T(t+1)$, $\cos(WD(t))$, $\cos(WD(t+1))$,$-\sin(WD(t))$, $-\sin(WD(t+1))$, $WS(t)$, $WS(t+1),R(t)$, $R(t+1)$, $H(t)$, $H(t+1)$
XC_{30}	WEI (t+1)

3.2. Selection of input variables using RF

We use 5-fold cross-validation error of the RF model to evaluate the different combinations of input variable candidates. The most important 15 variables are invoked in each iteration. In the first iteration, five candidates, including XC_{11}, XC_{15}, XC_{22}, XC_{28} and XC_{31}, are rejected. In the second iteration, another seven candidates (XC_1, XC_2,

XC_4, XC_{10}, XC_{14}, XC_{21} and XC_{27}) are rejected. No candidates can be rejected in the third iteration. The selected input variables are given in Table 2. Finally, we run the RF model to obtain the importance scores of the selected variables. As shown in Fig. 2, the variable X_{10} (the lagged 1-hour $PM_{2.5}$ concentration) is far more important than any other variable. The variable X_{15} (the lagged 1-hour wind speed) is the second important variable, which is followed by the variable X_{16} (the wind speed of the next hour). Let us remark that both XC_{26} and XC_{27} are rejected due to the small importance. Similarly in Poggi and Portier (2011), the daily total rainfall was not retained in the prediction model of the daily mean PM_{10} concentration.

Table 2: The selected input variables corresponding to the output $PM_{2.5}(t+1)$.

Input variable names	Input vector
$X_0,\ldots,X_{10}$	$CO(t)$, $NO_2(t-1)$,$NO_2(t)$,$O_3(t-2)$,$O_3(t-1)$,$O_3(t)$, $SO_2(t)$,$PM_{10}(t-2)$, $PM_{2.5}(t-2)$,$PM_{2.5}(t-1)$,$PM_{2.5}(t)$
$X_{11},\ldots, X_{18}$	$T(t)$, $T(t+1)$, -sin(WD(t)), -sin(WD($t+1$)), WS(t), WS($t+1$), H(t), H($t+1$)

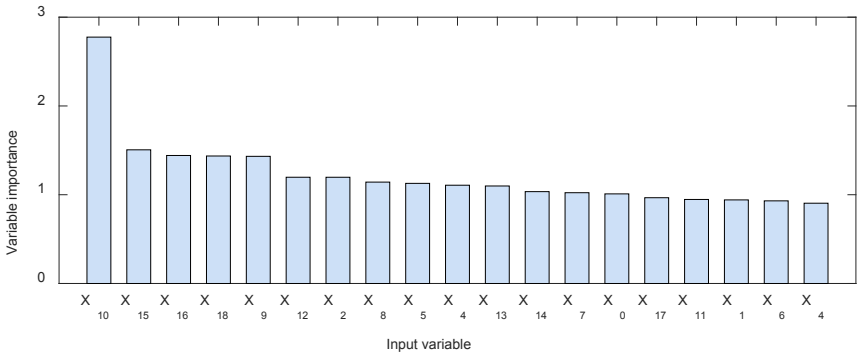

Fig. 2. Importance scores of the selected input variables

3.3. CART-EELM model

3.3.1. Splitting the dataset by using CART

This study constructs a shallow tree to capture concentration change patterns and ensure that each leaf has enough samples for its local model. To train a shallow regression tree, the minimum number of samples in a leaf and the maximum depth of the tree are set to 1000 and 3, respectively. In Fig. 3, the results of splitting the training set by using CART are given. The serial number for the root node is 0, and all the nodes choose X_{10} (the lagged 1-hour $PM_{2.5}$ concentration) to split its own dataset, since X_{10} contains the most information on next hour prediction (Lyu, et al, 2017). The persistence model is the simplest one, whose prediction for a given hour is the observed value of the previous hour.

The number of samples in nodes #3 and #4 reaches 10209, accounting for 52.02% of the total training set. Non-leaf nodes indicate the splitting rules. The value in each node is the mean value of the outputs of samples fallen into the node. There are significantly different MSE values for different leaves, which sharply increase as the serial numbers of the leaves increase.

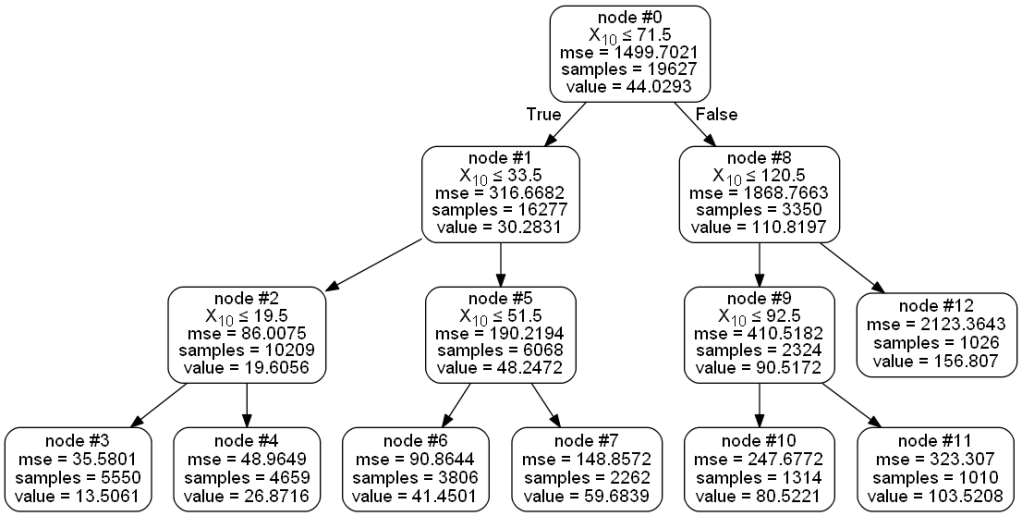

Fig. 3. Training results of the CART

3.3.2. Prediction model based on the CART–EELM

Each node in Fig. 3 has its own dataset. The hidden neuron numbers of EELM models are selected from $\{30,45,\ldots,300\}$. To seek the optimal numbers, 5-fold cross-validation is implemented. For each non-leaf node of the CART tree, EELMs are trained using the samples belonging to the node, and the hidden neuron numbers are selected to minimize validation errors respectively on the leaves of a sub-tree that takes the node as the root node. Before training EELMs, the input variables are normalized within [0,1] by

$$\overline{x}_i^d = \frac{x_i^d - \min(x_i^d \big|_{i=1}^{l})}{\max(x_i^d \big|_{i=1}^{l}) - \min(x_i^d \big|_{i=1}^{l})}, \tag{4}$$

where l denotes the number of samples, p the number of numerical variables, x_i^d the origin value of the dth number variable, and $\overline{x}_i^d$ the normalized value of the dth number variable.

The root node has 7 leaves. In order to show the selection procedure clearly, an example of training a global model at node #0 for node #3 is given. The total training set (the training set that belongs to node #0) is first divided into 5 separate subsets with approximately equal size. A series of 5 models are then trained, each using a different combination of 4 subsets. For each candidate of hidden neuron numbers, the validation error on node #3 is evaluated for each model, utilizing the subset of the data that are not used in training and belongs to node #3. Then the validation error is determined by averaging over all 5 validation subsets. The root mean square error (RMSE) is used as the error criterion, which is calculated by

$$\mathrm{RMSE} = \sqrt{\frac{1}{l}\sum_{i=1}^{l}(y_i - \hat{y}_i)^2}, \tag{5}$$

where $\hat{y}_i$ is the predicted value.

The first subplot in Fig. 4 demonstrates the average validation error on node #3 when using different candidates of hidden neurons numbers. The same selection procedure is implemented for other 6 leaves at node #0. The top first 7 subplots in Fig.3 show the validation errors of all global models at node #0. For the global models trained at node #0 for 7 leaf nodes, the optimal numbers of hidden nodes are 105, 255, 120, 90, 210, 300 and 135, respectively. It shows that different leaves may require different numbers of hidden neurons. The last subplot of Fig. 4 gives the validation errors of a global EELM on the total validation set. Its optimal model parameter is 135, which is resulted from balancing different leaf validation errors to minimize the total validation error.

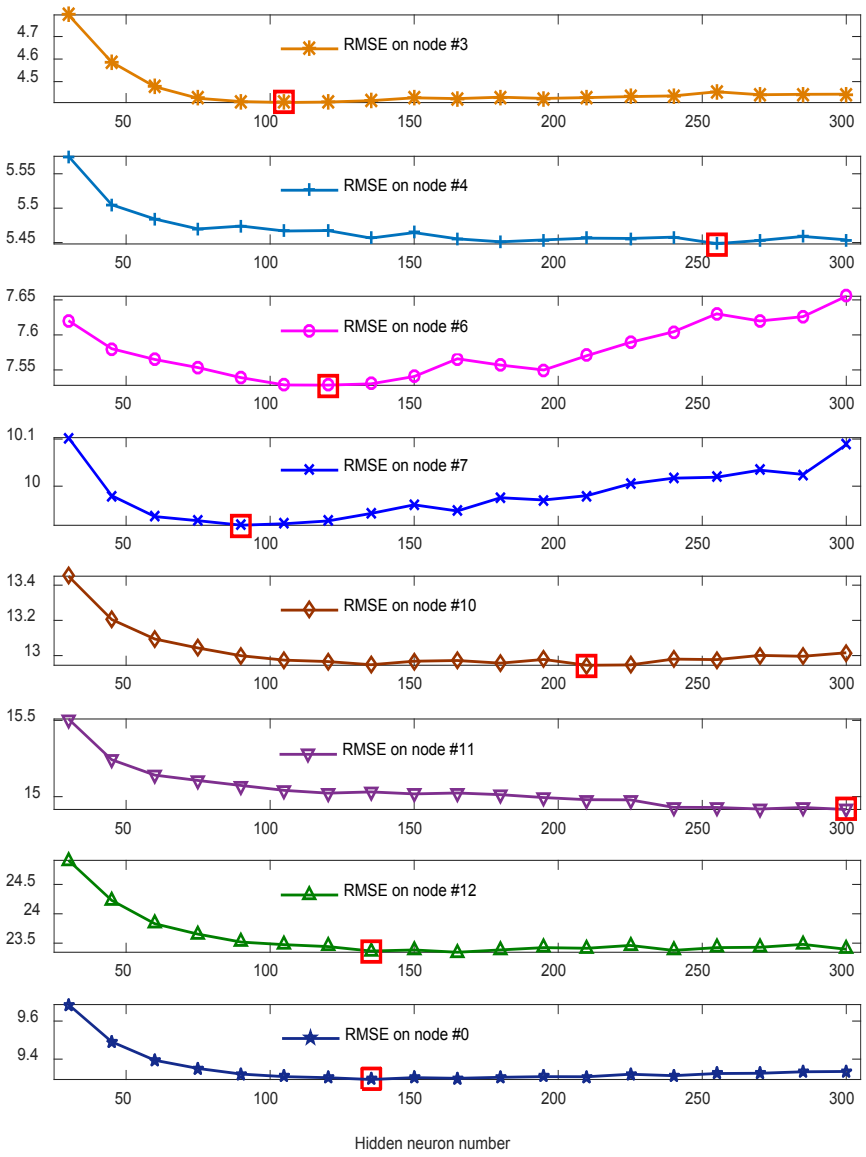

Fig. 4. Validation errors of the global models

Additional example of training local models is given at node #8 for nodes #10, #11 and #12. In Fig. 5, it shows the parameter selection results. For local models trained at node #8 for nodes #10, #11 and #12, the optimal numbers of hidden nodes are 60, 75 and 135, respectively.

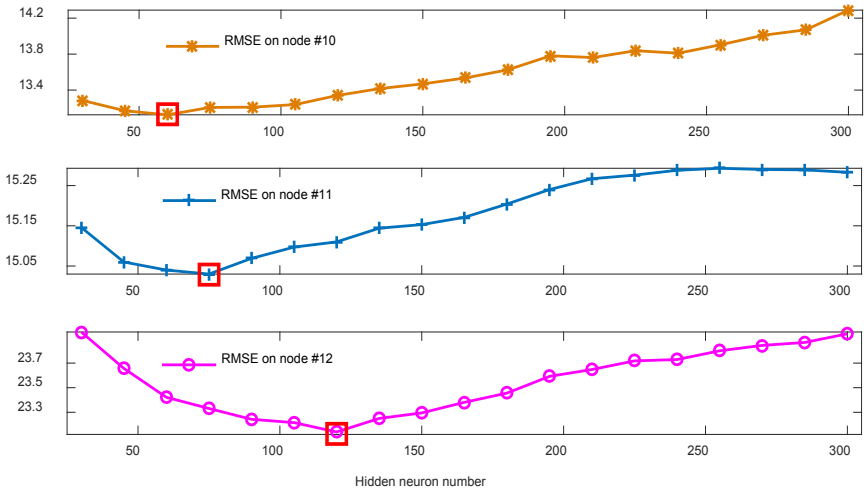

Fig. 5. Validation errors of local models at node #8

The models at all the nodes are organized in a hierarchical fashion, as shown in Fig. 6. $\mathrm{EELM}_{\#10}^{\#8}$, is shown as an example to clarify the information provided in the node. $\mathrm{EELM}_{\#10}^{\#8}$ represents the EELM model trained at node #8 for node #10. The optimal parameter is selected as 60 with the RMSE error of 13.13. For node #10 (a leaf), a global model and three local models, namely $\mathrm{EELM}_{\#10}^{\#0}$, $\mathrm{EELM}_{\#10}^{\#8}$, $\mathrm{EELM}_{\#10}^{\#9}$ and $\mathrm{EELM}_{\#10}^{\#10}$, are compared in terms of the RMSE error, and finally $\mathrm{EELM}_{\#10}^{\#0}$ is chosen. Consequently the prediction models for other 6 leaves are obtained.

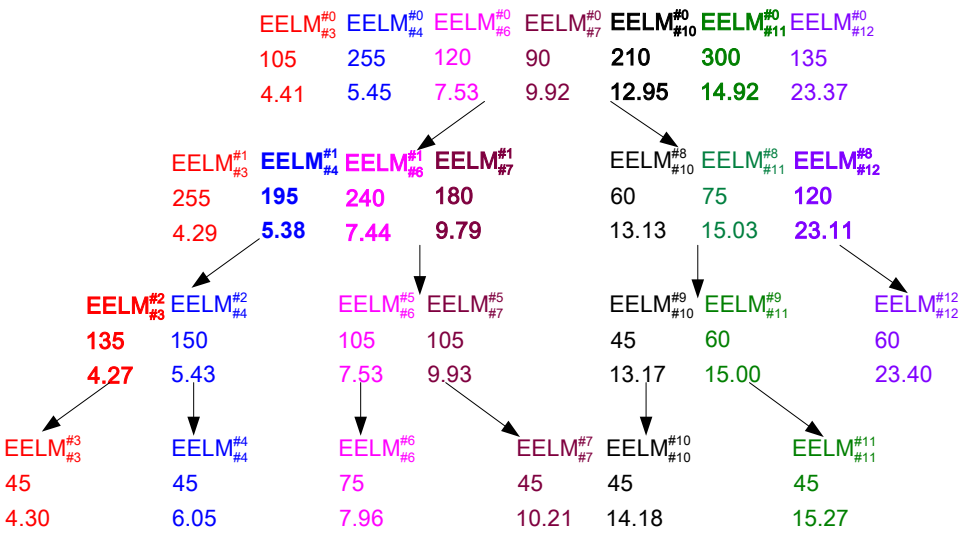

Fig. 6. EELM models at all nodes of the CART tree

For the input region of node #3, where X_{10} is not larger than 19.5μg·m^{-3}, the prediction model is trained at node #2, where X_{10} is not larger than 33.5μg·m^{-3}. Other three leaves in the left branch of the root node choose node #1, where X_{10} is not larger than 71.5μg·m^{-3}, to train the prediction models. For nodes #10 and #11, the prediction models are trained at the root node. The prediction model of node #12 is trained at node #8, where X_{10} is larger than 71.5μg/m^3. The selection procedure provides the flexibility to address the global-local duality.

3.4. Global models

In order to evaluate the model effectiveness, the model has been compared with the persistence model and global models, including RF, *v*-SVR and EELM. The model is also compared with local models based on season and clustering algorithm. In the RF model, it consists of 100 trees, and the model has a single parameter, i.e., the minimum number of samples in a leaf, which is selected from {1,2,…,8}. The value of ε in SVR chosen as a prior, is hard to determine. To overcome the difficulty of ε determination, the *v*-SVR (Schökopf et al., 2000) is used, whereby *v* controls the number of support vectors and training errors. In this study, Gaussian kernel function is applied in the *v*-SVR:

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2 / 2\gamma^2). \quad (6)$$

The *v*-SVR model owns three parameters: *v*, γ and the regularization parameter *C*, which are selected from {0.3,0.4,…,0.7},{0.1,0.15,…,0.45} and {0.5,1,…,3}, respectively. Thus in the *v*-SVR, there are 240 combinations of three parameter values. The parameter selection based on 5-fold cross-validation is also applied to several global models. The optimal parameter of the RF turns out to be 1. The optimized parameters of the *v*-SVR are selected as *v*=0.4, $\gamma = 0.45$ and $C = 1$. For the EELM, the optimized parameter is 135, which is indicated in the last subplot in Fig. 4.

3.5. Local models based on season and k-means

The seasonal models split the training set into three subsets based on prevailing weather patterns that may influence the $PM_{2.5}$ buildup. Fig. 7 presents the monthly mean temperatures, wind speeds and relative humilities as well as the monthly rainfalls over the period between July 1, 2015 and December 31, 2017. The monthly mean concentrations of $PM_{2.5}$ are indicated in Fig. 8. The $PM_{2.5}$ monthly mean concentration is negatively correlated to the monthly mean temperature and relative humidity as well as the monthly rainfall. The correlation coefficients are -0.88, -0.58 and -0.59, respectively.

A winter season model is trained for the time period between December and February, in which the monthly mean temperatures and the monthly rainfalls were generally low. As shown in Fig. 8, the monthly mean concentrations of $PM_{2.5}$ exceeded 60 $\mu g \cdot m^{-3}$ during this period. The data from March, April and November are used for the model corresponding to middle $PM_{2.5}$ levels. It is should be noted that in October 2015, high concentration episodes occurred, and the data from this month are used for middle $PM_{2.5}$ levels. Finally, the third model is trained for the time period between May and October. During this period, the monthly mean temperatures and the monthly rainfalls were generally high, and the monthly mean concentrations of $PM_{2.5}$ were lower than 40 $\mu g \cdot m^{-3}$.

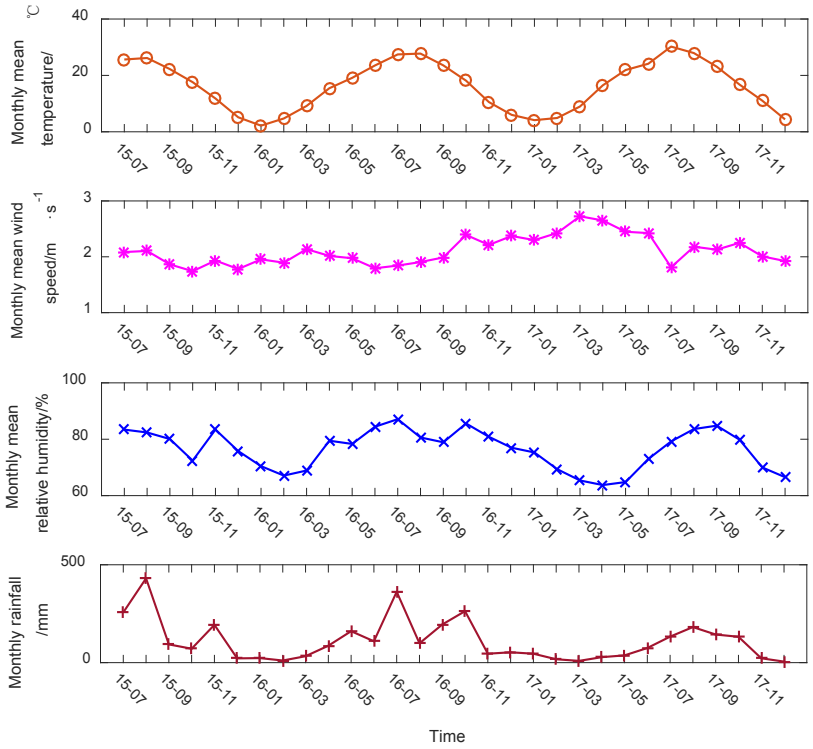

Fig. 7. Monthly mean concentrations from July, 2015 to December, 2017

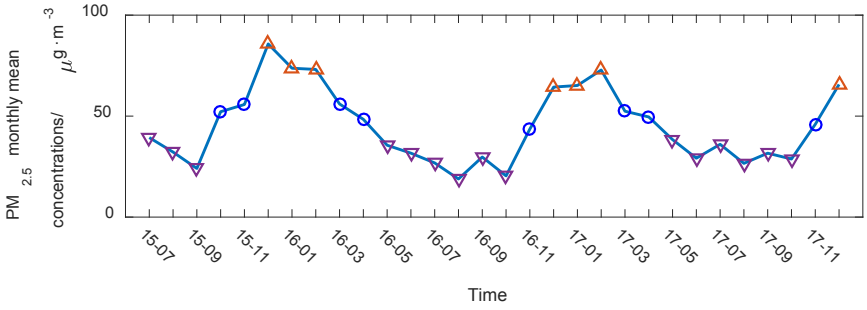

Fig. 8. Monthly mean concentrations of $PM_{2.5}$ from July, 2015 to December, 2017

The *k*-means clustering model aims to partition the *l* samples into *k* sets $\boldsymbol{S}=\{S_1,S_2,\ldots,S_k\}$ in such a way that within each cluster the average dissimilarity of the samples from the cluster mean is minimized. Mathematically, the objective of the *k*-means clustering is to find

$$\arg\min_{\boldsymbol{S}} \sum_{i=1}^{k} \sum_{x\in S_i} \left\| \boldsymbol{x} - \boldsymbol{\mu}_i \right\|^2 , \tag{7}$$

where $\boldsymbol{\mu}_i$ is the mean of points in S_i. Unfortunately, such optimization by complete enumeration is feasible only for very small data sets. The strategy based on iterative greedy descent is the most popular method to obtain a good suboptimal partition. In general, there are three steps involved in the strategy. Namely, 1. initialize *k* cluster centroids; 2. assign each training example to its closest centroid; 3. recalculate each cluster centroid using the examples assigned to it in Step 2, and go back to Step 2 until convergence. The algorithm repeats Step 2 and Step 3 until the centroids do not change between two consecutive iterations.

The prediction method based on *k*-means divides the dataset into several clusters using *k*-means and builds a local EELM for individual cluster. The number of clusters is critical to the performance of the local models. The value of *k* can be determined by various methods. In this study, 5-fold cross-validation is used to choose the optimal *k* ranging from 2 to 7. Firstly, the total training set is partitioned into *k* clusters. Secondly in each cluster, we train a local EELM, whose hidden neuron number is selected by 5-fold cross-validation. Then for a certain *k*, the validation error averaged over different clusters is obtained. Finally, the validation errors for different values of *k* are compared, and the one with the minimum validation error is chosen. In the study, the optimal number of clusters turns out to be 3.

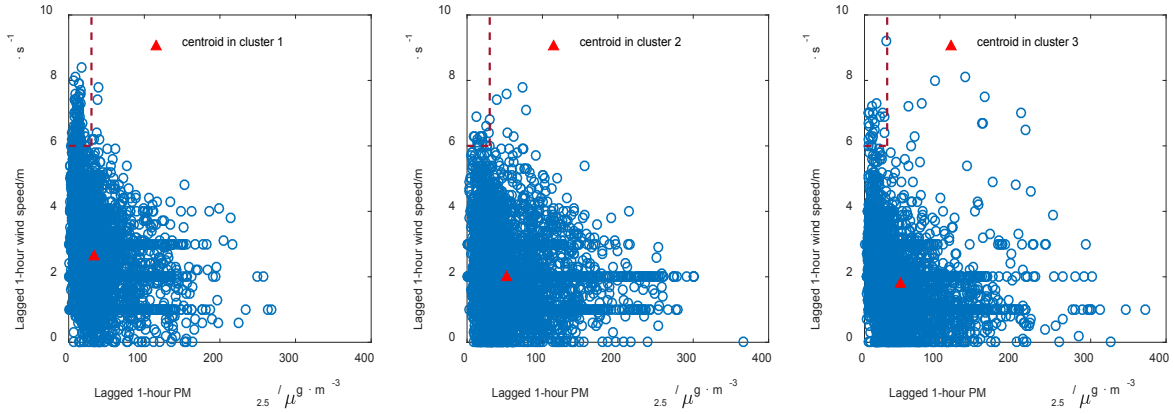

Fig. 9. Scatter plots between the lagged 1-hour $PM_{2.5}$ concentration and the lagged 1-hour wind speed in three clusters

To extract the characteristics in three clusters, the scatter plots between the lagged 1-hour $PM_{2.5}$ concentration and the lagged 1-hour wind speed in three clusters are given in Fig. 9, since these two variables are the two most important predictors. In Fig. 9, cluster 1 tends to include more samples that have low lagged 1-hour $PM_{2.5}$ (less than 30μg·m^{-3}) but high lagged 1-hour wind speed (higher than 6 m·s^{-1}). In three clusters, the two most important elements of the centroids are (33.17, 2.61), (52.47, 1.98) and (47.17, 1.78), respectively. It may be unreasonable that the latter two are closely spaced.

3.6. Prediction results and discussion

The training times required by different models are given in Table 3. The RF model has the shortest training time, while the *v*-SVR has the highest computational cost, since the *v*-SVR needs to solve a quadratic programming problem for each combination of parameter values. The training time of the *k*-means-EELM is more than four times that of the EELM. Due to high computational efficiency of the EELM, the training times of three kinds of local models are significantly shorter than that of the *v*-SVR.

The testing errors from different models are also given in Table 3. The testing set is split by the CART in Fig. 3. The errors in seven input regions, corresponding to the seven leaves, are also included in Table 3. The persistence model services as a benchmark for other models. Except in the nodes #3 and #4, the prediction

accuracy of the RF is inferior to those of the *v*-SVR and the EELM. Splitting the training set based on season increases the performance of the EELM, but the *k*-means-EELM is slightly inferior to the EELM in terms of accuracy on the total testing set.

Table 3: Testing results of different models.

Models	Training time/m	Testing RMSE/μg·m^{-3}	Testing RMSEs on the leaves/μg·m^{-3}						
			#3	#4	#6	#7	#10	#11	#12
Persistence	-	10.10	4.99	6.23	8.25	10.44	12.34	15.71	20.01
RF	**23.08**	9.81	**4.61**	6.07	8.02	9.58	11.53	14.63	20.87
v-SVR	1201.12	9.23	4.80	6.11	7.94	9.49	**10.78**	13.52	18.25
EELM	38.03	9.22	4.82	6.10	7.92	9.57	10.97	13.30	18.10
Seasonal EELM	35.42	9.12	4.86	6.06	7.88	9.45	10.86	13.28	17.69
k-means-EELM	161.23	9.25	4.86	6.19	7.94	9.61	10.95	13.33	18.13
CART-EELM	87.05	**8.96**	4.63	**6.04**	**7.86**	**9.35**	10.82	**13.20**	**17.02**

For the results, the CART-EELM shows the most satisfactory accuracy on the total testing set. At leaves #4, #6 and #7, the CART-EELM achieves the best prediction accuracy. At node #10, the prediction performance of CART-EELM is slightly worse than the *v*-SVR. At node #12, the prediction error of the CART-EELM drops by 5.97% and 3.79% compared with the EELM and the seasonal EELM, respectively. To further compare the CART-EELM with other models, the mean absolute error (MAE), the mean absolute percentage error (MAPE) and correlation coefficient the on the total testing set are given in Table 4. The MAE and the MAPE are calculated by

$$\mathrm{MAE} = \frac{1}{l}\sum_{i=1}^{l}\left|y_i - \hat{y}_i\right|, \tag{8}$$

$$\mathrm{MAPE} = \frac{1}{l}\sum_{i=1}^{l}\left|\frac{y_i - \hat{y}_i}{y_i}\right|. \tag{9}$$

The CART-EELM model has the lowest testing MAE and MAPE on the total testing set. It is found that the correlation coefficient has the lowest sensitivity. Among these statistical indicators, the RMSE is proposed as the key one for the description of the model skill (Thunis et al., 2012).

Table 4: Other statistical indicators of different models.

Models	Testing MAE/μg·m^{-3}	Testing MAPE/%	correlation coefficient
Persistence	6.43	16.85	0.97
RF	6.24	16.76	0.97
v-SVR	6.04	16.78	0.98
EELM	5.99	16.46	0.98
Seasonal EELM	5.99	16.51	0.98
k-means-EELM	6.04	16.71	0.98
CART-EELM	**5.86**	**16.01**	0.98

According to the newly updated Ambient Air Quality Standard (GB3095-2012) and combined with the actual air pollution condition of three Chinese cities, the ambient air quality levels and the corresponding concentration ranges of $PM_{2.5}$ were developed (Xu et al., 2017). Table 5 presents the levels and the corresponding ranges. The correct estimations, the underestimates and the overestimates are counted for the EELM, the seasonal EELM and the CART-EELM. Compared with an overestimation, an underestimation may bring more harm to the public. Statistical results of three models are indicated in Table 6. The first row of the CART-EELM, is taken as an example to clarify the information provided in Table 6. There are 437 samples with the actual levels being Ⅰ.

Among these, 279 samples are correctly estimated and the rest 158 samples are overestimated as level Ⅱ. From Table 6, it is found that the CART-EELM has the most correctly estimated samples and the least underestimated samples.

Table 5: The ambient air quality levels and the corresponding concentration ranges

Levels				
Good (Ⅰ)	Regular (Ⅱ)	Bad (Ⅲ)	Very bad (Ⅳ)	Extreme bad (Ⅴ)
(0,15]	(15,35]	(35,75]	(75,120]	(120,+∞)

Table 6: Statistical results of the estimated levels from the EELM, the seasonal EELM and the CART-EELM

Actual	EELM					Seasonal EELM					CART- EELM				
levels	Ⅰ	Ⅱ	Ⅲ	Ⅳ	Ⅴ	Ⅰ	Ⅱ	Ⅲ	Ⅳ	Ⅴ	Ⅰ	Ⅱ	Ⅲ	Ⅳ	Ⅴ
Ⅰ	280	157	0	0	0	280	157	0	0	0	279	158	0	0	0
Ⅱ	77	1214	130	1	0	85	1212	124	1	0	70	1223	129	0	0
Ⅲ	2	104	1068	62	0	1	104	1066	65	0	1	103	1071	61	0
Ⅳ	0	0	79	414	27	0	0	78	417	25	0	0	69	423	28
Ⅴ	0	0	0	38	245	0	0	1	39	243	0	0	0	35	248

In this study, the CART-EELM model has better performances compared to the other models discussed. The global model trained on the total dataset is hard to fit well all the patterns, and the local models trained at the leaves have the overfitting risk. The current model presented aims to deal with the global-local duality for each leaf. The prediction model for a leaf is selected among a global and several local EELMs on the path from the root node to the leaf, in terms of the validation error on the leaf. The selection procedure provides the CART-EELM the flexibility to deal with the global-local duality.

The evaluation results also show that splitting the dataset based on season increases the prediction accuracy of the EELM model. Compared to the global model based on the EELM, local models based on *k*-means cannot improve the performance. In the case, the CART-EELM model shows better effectiveness of the $PM_{2.5}$ concentration prediction.

4. Conclusions

The multiple change patterns of $PM_{2.5}$ concentrations increase the difficulty of hourly $PM_{2.5}$ prediction. As local models show great potential to improve the prediction accuracy, local prediction models based on the CART-EELM is proposed, which uses the CART to split the training set into subsets in the fashion of a hierarchical tree and EELMs are trained at each node with its own training samples. The prediction model for each leaf is then selected among a global model and local models on the path from the root node to the leaf in terms of the validation error on the leaf.

For the experimental results of evaluation, it can be concluded: (1) the method can address the global-local duality for the prediction model on each leaf; (2) the CART-EELM method has better performances than the global models, including RF, *v*-SVR and EELM; (3) the CART-EELM method also show outperforms compared to the seasonal EELM and the *k*-means-EELM.

Acknowledgements

This research was supported in part by the research fund of Key Laboratory for Advanced Technology in Environmental Protection of Jiangsu Province (Grant No. AE201121).

References

Bai, Y., Li, Y., Wang, X.X., Xie, J.J., Li, C., 2016. Air pollutants concentrations forecasting using back propagation neural network based on wavelet decomposition with meteorological conditions. Atmos. Pollut. Res. 7, 557–566.

Bettenburg, N., Nagappan, M., Hassan, A.E., 2012. Think locally, act globally: Improving defect and effort prediction models. IEEE International Working Conference on Mining Software Repositories, 60–69.

Breiman, L., Friedman, J.H., Olshen, R., Stone, C.J., 1984. Classification and regression trees. Biometrics 40, 358.

Breiman, L., 2001. Random forests. Mach. Learn. 45, 5–32.

Brezigar-Masten, A., Masten, I., 2012. CART-based selection of bankruptcy predictors for the logit model. Expert Syst. Appl. 39 (11), 10153–10159.

Budalakoti, S., Srivastava, A.N., Otey, M.E., 2009. Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety. IEEE Trans. Syst. Man Cybern. C Appl. Rev. 39 (1), 101–113.

Cobourn, W.G., (2010). An enhanced $PM_{2.5}$ air quality forecast model based on nonlinear regression and back-trajectory concentrations. Atmos. Environ. 44 (25), 3015–3023.

Díaz-Robles, L.A., Ortega, J.C., Fu, J.S., Reed, G.D., Chow, J.C., Watson, J.G., et al., 2008. A hybrid ARIMA and artificial neural networks model to forecast particulate matter in urban areas: the case of Temuco, Chile. Atmos. Environ. 42, 8331–8340.

Dong, M., Yang, D., Kuang, Y., He D., Erdal S., Kenski D., 2009. $PM_{2.5}$ concentration prediction using hidden semi-Markov model-based times series data mining. Expert Syst. Appl. 36 (5), 9046–9055.

Doraiswamy, P., Hogrefe, C., Hao, W., Civerolo, K., Ku, J.Y., Sistla, G., 2010. A retrospective comparison of model-based forecasted $PM_{2.5}$ concentrations with measurements. J. Air & Waste Manage. Assoc. 60, 1293–1308.

Feng, Y., Zhang, W., Sun, D., Zhang, L., 2011. Ozone concentration forecast method based on genetic algorithm optimized back propagation neural networks and support vector machine data classification. Atmos. Environ. 45, 1979–1985.

Feng, X., Li, Q., Zhu, Y.J., Hou, J.X., Jin, L.Y., Wang, J.J., 2015. Artificial neural networks forecasting of $PM_{2.5}$ pollution using air mass trajectory based geographic model and wavelet transformation. Atmos. Environ. 107, 118–128.

Garcia, J.M., Teodoro, F., Cerdeira, R., Coelho, L.M.R., Kumar, P., Carvalho, M.G., 2016. Developing a methodology to predict PM10 concentrations in urban areas using generalized linear models, Environ. Technol., 37(18), 2316–2325.

He, J.J., Gong, S.L., Yu, Y., Yu, L.J., Wu, L., Mao, H.J., et al., 2017. Air pollution characteristics and their relation to meteorological conditions during 2014–2015 in major Chinese cities. Environ. Pollut. 223, 484–496.

Huang, G.B., Zhu. Q,Y., Siew, C.K., 2006. Extreme learning machine: theory and applications. Neurocomputing 70 (1-3), 489–501.

Kassomenos, P.A., Vardoulakis, S., Chaloulakou, A., Paschalidou, A.K., Grivas, G., Borge, R., et al., 2014. Study of PM_{10} and $PM_{2.5}$ levels in three European cities: Analysis of intra and inter urban variations. Atmos. Environ. 87, 153–163.

Khoshgoftaar, T.M., Hulse, J.V., Napolitano, A., 2011. Comparing Boosting and Bagging techniques with noisy and imbalanced data, IEEE Trans. Syst. Man Cybern. A Syst. Humans 41 (3), 552–568.

Liu, Z.R., Hu, B., Wang L.L., Wu, F.K., Gao, W.K., Wang, Y.S., 2015. Seasonal and diurnal variation in particulate matter (PM_{10} and $PM_{2.5}$) at an urban site of Beijing: analyses from a 9-year study. Environ. Sci. Pollut. Res. 22 (1), 627–642.

Lu, W.Z., Wang, W.J., 2005. Potential assessment of the "support vector machine" method in forecasting ambient air pollutant trends. Chemosphere, 59, 693–701.

Lyu, B.L., Zhang, Y.Z., Hu, Y.T., 2017. Improving $PM_{2.5}$ air quality model forecasts in China using a bias-correction framework. Atmosphere 8, 147.

McKendry, I.G., 2002. Evaluation of artificial neural networks for fine particulate pollution. J. Air & Waste Manage. Assoc. 52 (9), 1096–1101.

Ni, X.Y., Huang, H., Du, W.P., 2017. Relevance analysis and short-term prediction of $PM_{2.5}$ concentrations in Beijing based on multi-source data. Atmos. Environ. 150, 146–161.

Niu, M.F., Wang, Y.F., Sun, S.L., Li, Y.W., 2016. A novel hybrid decomposition-and-ensemble model based on CEEMD and GWO for short-term $PM_{2.5}$ concentration forecasting. Atmos. Environ. 134, 168–180.

Osowski, S., Garanty, K., 2007. Forecasting of the daily meteorological pollution using wavelets and support vector machine. Eng. Appl. Artif. Intel. 20, 745–755.

Perez, P., 2012. Combined model for PM_{10} forecasting in a large city. Atmos. Environ. 60, 271–276.

Perez, P., Gramsch, E., 2016. Forecasting hourly $PM_{2.5}$ in Santiago de Chile with emphasis on night episodes. Atmos. Environ. 124, 22–27.

Poggi, J.M., Portier, B., 2011. PM10 forecasting using clusterwise regression. Atmos. Environ. 45, 7005–7014.

Qin, S.S., Liu, F., Wang, J.Z., Sun, B.B., 2014. Analysis and forecasting of the particulate matter (PM) concentration levels over four major cities of China using hybrid models. Atmos. Environ. 98, 665–675.

Requia, W.J., Adams M.D., Koutrakis, P., 2017. Association of $PM_{2.5}$ with diabetes, asthma, and high blood pressure incidence in Canada: A spatiotemporal analysis of the impacts of the energy generation and fuel sales. Sci. Total Environ. 584-595, 1077–1083.

Russo, A., Raischel, F., Lind, P.G., 2013. Air quality prediction using optimal neural networks with stochastic variables. Atmos. Environ. 79, 822–830.

Schölkopf, B., Smola, A.J., Williamson, R.C., Bartlett, P.L., 2000. New support vector algorithms. Neural Comput. 12 (5), 1207–1245.

Sun,W., Zhang, H., Palazoglu, A., Singh, A., Zhang, W.D., Liu, S.W., 2013. Prediction of 24-hour-average $PM_{2.5}$ concentrations using a hidden Markov model with different emission distributions in Northern California. Sci. Total Environ. 443, 93–103.

Thomaidis, N.S., Bakeas, E.B., Siskos, P.A., 2003. Characterization of lead, cadmium, arsenic and nickel in $PM_{2.5}$ particles in the Athens atmosphere, Greece. Chemosphere 52 (6), 959–966.

Thunis, P., Pederzoli, A., Pernigotti, D., 2012. Performance criteria to evaluate air quality modeling applications. Atmos. Environ. 59, 476–482.

Vlachogianni, A., Kassomenos, P., Karppinen, A., Karakitsios, S., Kukkonen, J., 2011. Evaluation of a multiple regression model for the forecasting of the concentrations of NO_x and PM_{10} in Athens and Helsinki. Sci. Total Environ. 409 (8) , 1559–1571.

Voukantsis, D. Karatzas, K., Kukkonen, J., Räsänen, T., Karppinen, A., Kolehmainen, M., 2011. Intercomparison of air

529 quality data using principal component analysis, and forecasting of PM_{10} and $PM_{2.5}$ concentrations using artificial
530 neural networks, in Thessaloniki and Helsinki. Sci. Total Environ. 409, 1266–1276.
531 Wang, D.Y., Wei, S., Luo H.Y., Yue, C.Q., Grunder, O., 2017. A novel hybrid model for air quality index forecasting
532 based on two-phase decomposition technique and modified extreme learning machine. Sci. Total Environ. 580,
533 719–733.
534 Xu, Y.Z., Yang, W.D., Wang, J.Z., 2017. Air quality early-warning system for cities in China. Atmos. Environ. 148,
535 239–257.
536 Xue, X.W., Yao, M., Wu, Z.H., Yang, J.H., 2014. Genetic ensemble of extreme learning machine. Neurocomputing 129,
537 175–184.
538 Yu, R.Y., Yang, Y., Yang, L.Y., Han, G.J., Move, O.A., 2016. RAQ-A random forest approach for predicting air
539 quality in urban sensing systems. Sensors 16 (1), 86.
540 Zhou, Z.H., Wu, J.S., Tang, W., 2002. Ensembling neural networks: Many could be better than all. Artif. Intell. 137,
541 239–263.
542 Zhou, Q.P., Jiang, H.Y., Wang, J.Z., Zhou, J.L., 2014. A hybrid model for $PM_{2.5}$ forecasting based on ensemble
543 empirical mode decomposition and a general regression neural network. Sci. Total Environ. 496, 264–274.
544 Zhu, S.L., Lian, X.Y., Liu, H.X., Hu, J.M., Wang, Y.Y., Che, J.X., 2017. Daily air quality index forecasting with hybrid
545 models: A case in China. Environ. Pollut. 231, 1232–1244.